Final Project Lab Report

Jibin Mathews., R# 11577748

Group 16: James Panhorst, Joshua Ramirez, Alex Holmes

Texas Tech University

November 2021

Abstract

The main project entails a rover that is able to follow a metallic strip path autonomously while also being able to overcome a corner ramp. Somewhere on the path an object lies which the rover has to detect and take an alternative path if it is available. The rover must be able to diverge and converge to and from the alternative path.

Table of Contents

List of Figures

1. Introduction

The initial approach to this project involved using standard coding procedures. However, we soon discovered that for a crucial portion of the project, state machines needed to be used. In this case, a new flow chart had to be implemented.

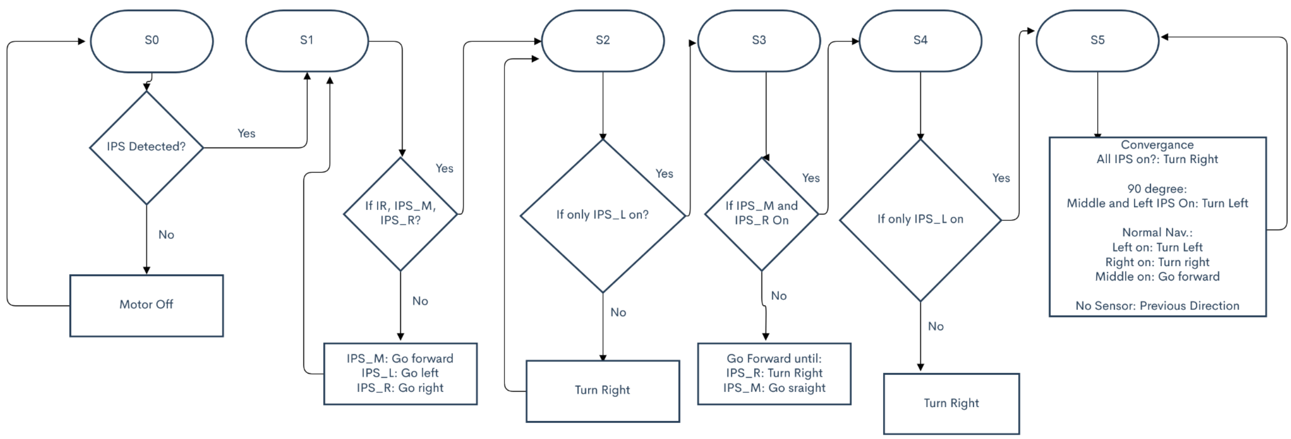


Figure 1: State Machine Flowchart. [1]

By using state machines, the coder can set different actions depending on what state the code is in. Based on this flowchart, the project was able to be coded while also thinking about the general logic diagram of the project. From this flowchart the project can be broken down into smaller portions that can be addressed individually. The project utilized the normal functioning of IPS and IR to navigate the path normally.

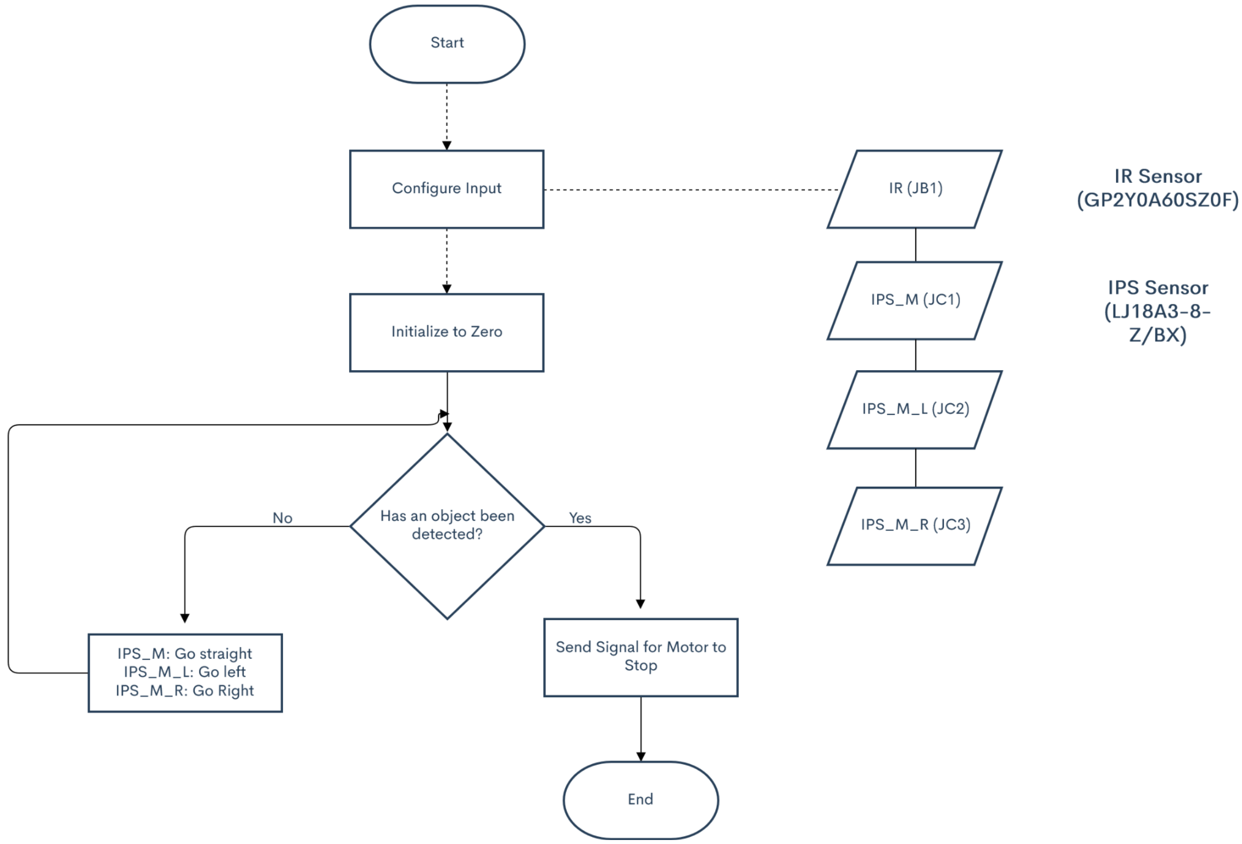


Figure 2: IPS and IR Flowchart.

The IPS and IR flow chart [Figure 2] provides a general idea of how the rover is intended to behave. Initially, all the inputs are configured such as the IR and the three IPS sensors used. The inputs are then initialized to zero. The parent code function determines whether there is an object present in front of the rover. Depending on this input the rover can either continue along the path or not move. The color sensor would be the main priority element of the code since the process requires an input of Red, green or Blue to continue with the corresponding step.

2. Components

Multiple components were used in order to achieve the objective. Specifically, the voltage regulator, comparator, optocoupler, Inductive Proximity Sensor, Infrared sensor, Light to Frequency sensor, Basys board and H-bridge. The Basys board has a limit for the maximum voltage of 5.5 volts direct current (DC) The voltage regulator (LM340T5) [Figure 3] was used in order to connect the voltage source of the 9.6 Volt battery to the basys board.

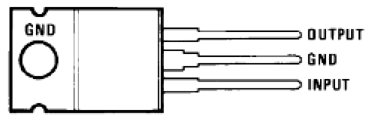


Figure 3: LM340T5 Voltage Regulator [2].

The regulator takes the input and uses a closed feedback loop in order to maintain a constant voltage for the output. The output is then directed to the basys board to provide a constant 5 Volts. Using a voltage regulator in order to connect the battery power supply to the basys board proved to be efficient and cost effective. In order to provide power for the motor and change the direction of current flow, an H-Bridge is required (L298) [Figure 4]. This component was used in order for the rover to make turns while navigating the path from small adjustments to 90 degree turns.

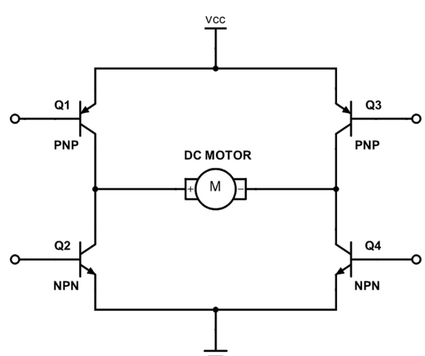


Figure 4: L298 H-Bridge [3].

The Inductive Proximity Sensor (IPS) [Figure 6] was used in order to identify the metal tape that is on the track. With an 8 millimeter (mm) detection range, the IPS sits in different locations on the rover in order to make sure it is correctly following the line. The specific Proximity sensor used is the LJ1A3-8-Z/BX which was received from the stockroom. In order to understand how the sensor is able to detect metal the schematic of the sensor should be understood.

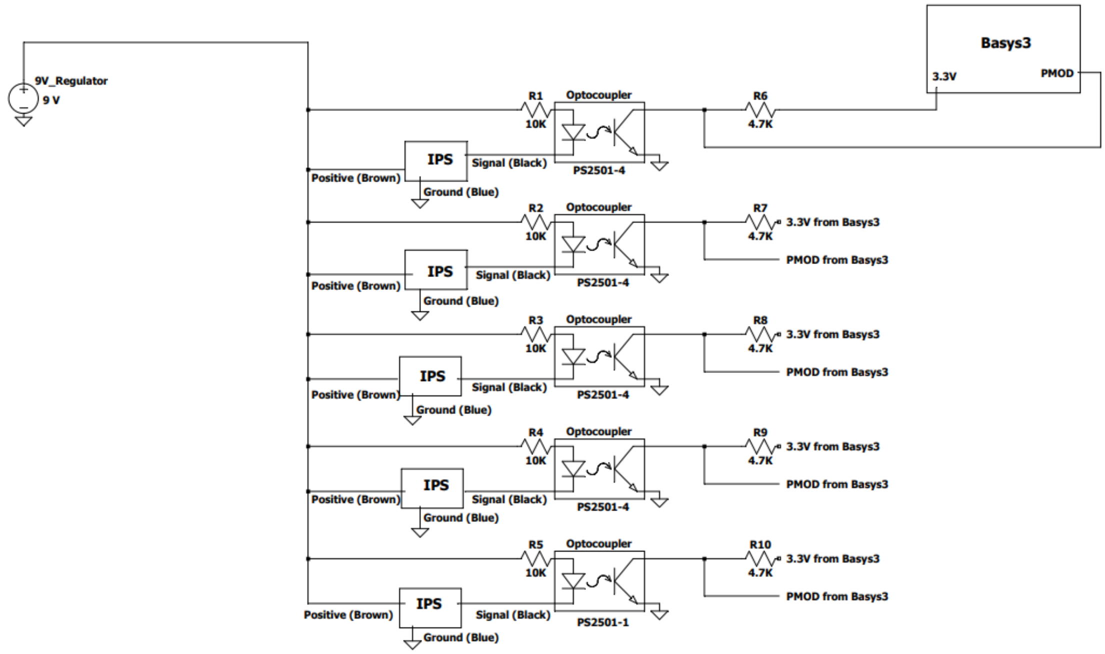


Figure 5: IPS Schematic [4].

Figure 6: LJ18A3-8-Z/BX Inductive Proximity Sensor.

The colors on the schematic represent the different wires that are available from the IPS [Figure 5]. The Brown wire signifies the positive lead, which is powered by a 9 volt regulator. The blue wire represents the ground and the black wire used as the signal wire. Within the IPS, a Bipolar Junction Transistor is used with a negative switching signal. NPN type resistors are on naturally which means whenever an object is present the signal is measured as high. While there is no object present the signal is sent low (Cook, 2021).

Within the IPS circuit an optocoupler was used in order to plug into the basys board while also maintaining a voltage that is compatible. The optocoupler [Figure 7] is used in order to transfer electrical signals through a light emitting diode often used for small circuits which require a low voltage (Jameco, n.d.). From there, a pullup resistor is used which is connected to a 3.3 Volt (DC) source. This resistor allows the outputs to be either 0 Volts for when no object is present and 3.3 Volts when an object is present. These results can be changed through the code by using an inverting function if the opposite case is required. By utilizing the optocoupler and IPS wires, a printed circuit board [Figure 7] was implemented to gain experience on the use of EagleCad and also to compress the circuit for the IPS sensors connection to the Basys board.

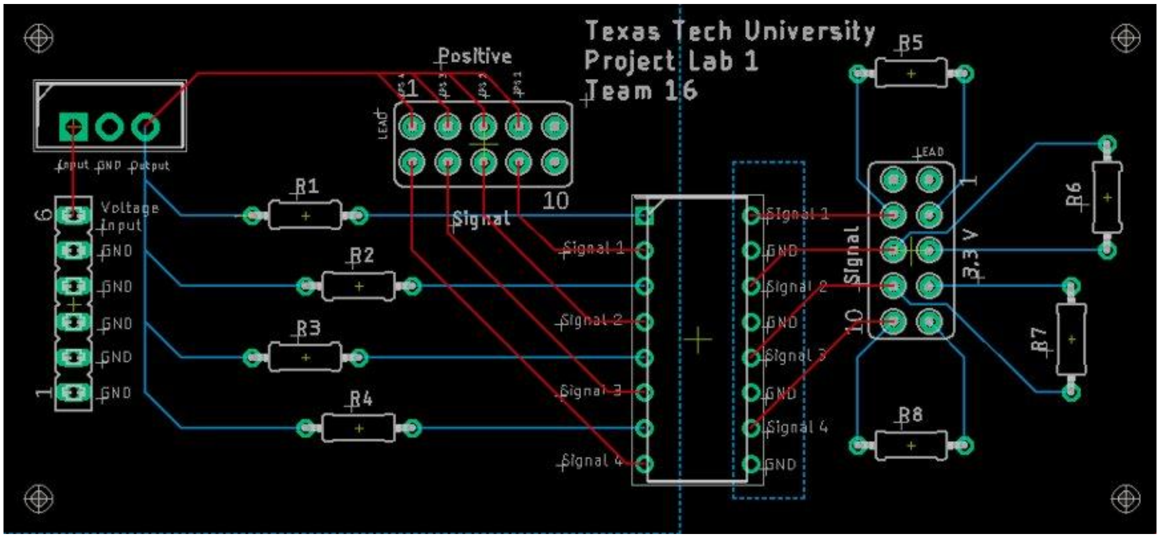


Figure 7: IPS PCB [5]

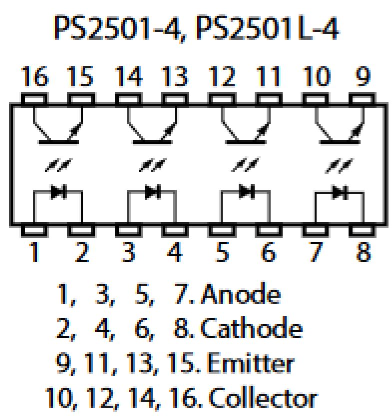


Figure 8: PS2501-4 Optocoupler [6].

Another important component utilized in the project is the comparator (LM399) [Figure 4]. The comparator is used in order to implement the software overcurrent protection and is also used to implement the IR sensor to be compared with a set value which depicts the presence of an object. The comparator essentially takes two values which are compared and outputs a digital signal of the larger signal (Basic Electronic Tutorials, 2021).

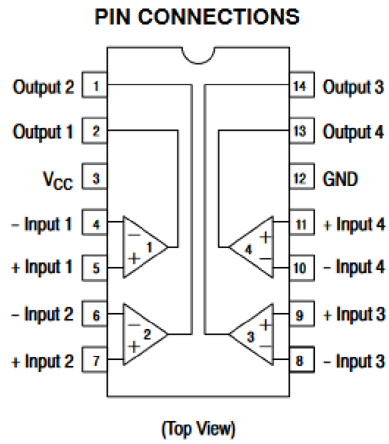


Figure 9: LM339 Quad-Comparator [7].

The specific comparator used contained four different mini comparators within the component. However, only two were required in order to compare the sensor from motor A and sensor from motor B. These sensors send a voltage to their corresponding comparator which is compared to 1 volt coming from the non- inverted portion. Even though the software overcurrent was implemented, a 2 Amp (Ampere) ATM fuse was also connected to the positive side of the battery to provide extra precaution from providing too much current in the event of a mishap.

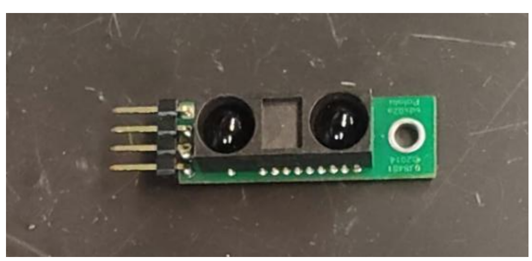


Figure 10: IR Sensor.

The Infrared Sensor (IR) (GP2Y0A60SZLF) uses a light transponder and receiver in order to determine the distance from an object that is placed in front of the sensor [Figure 9]. The specific IR sensor has a detecting range from 10 centimeters (cm) to 150 centimeters (cm). The diagram is set to have a 5 Volt input connected to Vcc directly to the IR sensor. The output is then sent to the comparator which compares the value to a 5 Volt input from the power source. This comparison is essential in order to determine if an object is present. The voltage divider with R2 and R3 are used to set the range of the IR sensors which in turn causes the IR sensor readings to be within the specified range. [Figure 10] Even though the resistors can be adjusted to set the detecting range, during demo day the object could be placed anywhere along the bath between the diverging and converging path so the best case possible was setting the object approximately 4 to 6 inches.

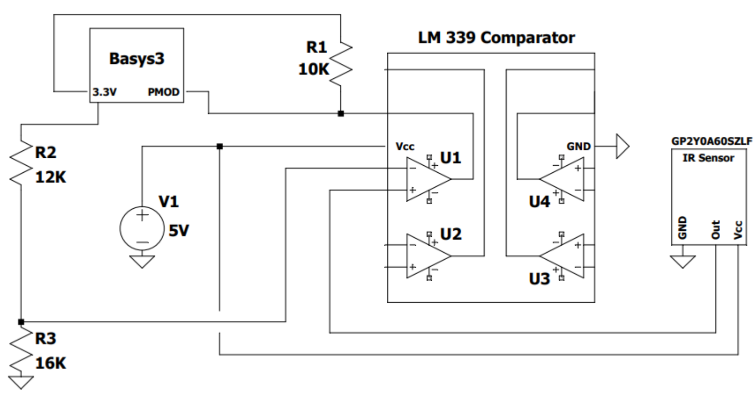


Figure 11: IR Schematic [8].

The color sensor is also used in order to determine different functioning of the rover. For example, green should make the rover go, blue should make the rover slowdown, and red should make the rover stop. The specific color sensor used for this project is the TCS 3200. The more defined term for this component is the Light to Frequency converter. The coding process involved multiple calculations that needed to be carried out. Overall, the sensor was able to detect and display on the Basys Board through LED's of the corresponding color however, the actual implementation to the rover was unsuccessful.

3. Hardware Setup

After defining and understanding the components involved within the project, the components can be combined and connected to each other in order to finalize the hardware portion of the project. Once all the hardware and software have been implemented, the final step involves wire management in order to provide aesthetics and also be easy to access for troubleshooting.

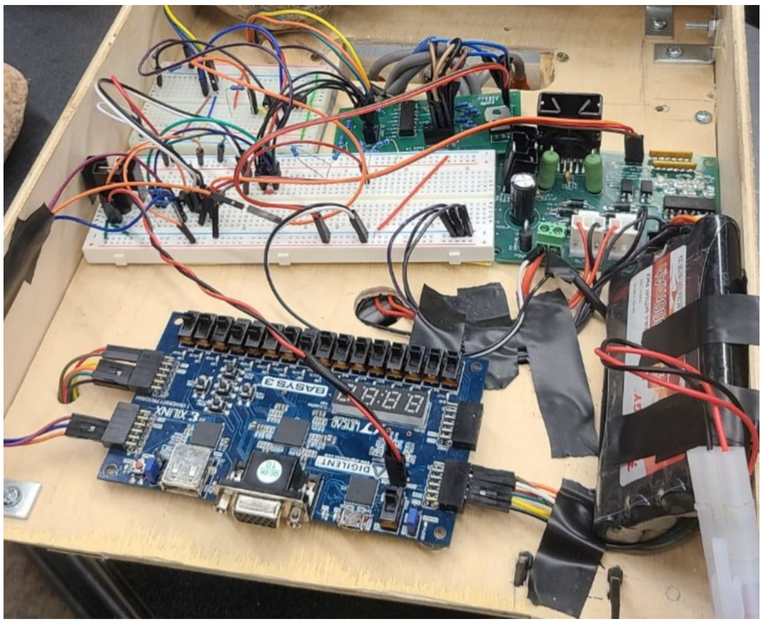


Figure 12: Final Placement and Wiring.

The wires present [Figure 11] are used to connect the basys board to power and the set up for overcurrent protection. Initially, electrical tape was used in order to keep wires in place. However, the tape was not adhesive enough to maintain the wires. The final touch ups involved utilizing duct tape in order to properly secure the wires. Zip ties were also used to secure the board and battery in order to survive the shake test.

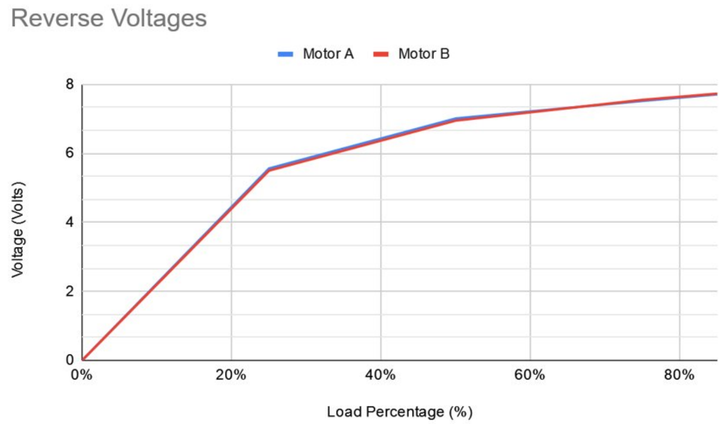
4. Software Flowchart

The development of the code can be simplified into different portions [Figure 1]: the color sensor, IPS, and IR sensor. The color sensor, which was not implemented, was meant to indicate when the rover should stop, go and slow down. The code that was developed used LED's to indicate which color was being detected. The IPS was needed in order to navigate the track normally while the IR was used to detect an obstacle. A major portion of the project involved the rover being able to identify an object and take an alternative path if available. In order to implement the coding for this requirement state machines needed to be used which helps with automation of the rover. The code is set up to detect an input from the IR and continuing until both the middle and right IPS were active. Once this occurs the rover is put into a different state to make a right turn onto the alternative path.

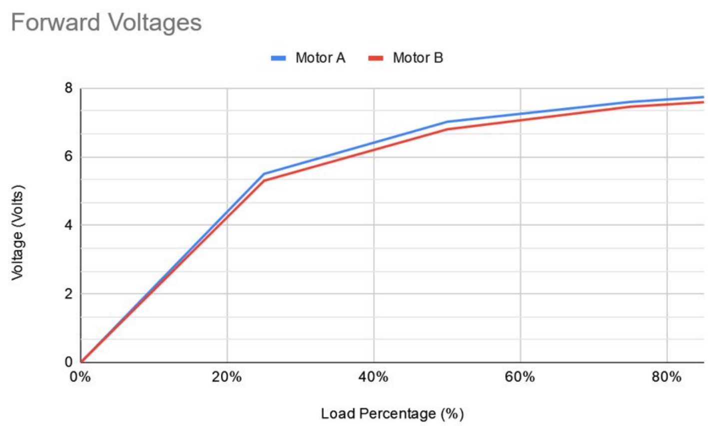
```
always @(posedge clock) begin
 case(state)
   S0: begin // Case state 0
     if (~IPS_M || ~IPS_M_L || ~IPS_M_R) state <= S1; // if any IPS detects tape, go to state 1
     else    state <= S0; // if no IPS is detected, stay in state 0
   end
   S1: begin // travels the track normally
//         if (IR && ~IPS_R_R) state <= S2;
         if (IR && ~IPS_M && ~IPS_M_R) state <= S2;
       else    state <= S1;
   end
   S2: begin  //only turns left
//       if (~IPS_M_L && IPS_M && IPS_M_R && IPS_R_R) state <= S3;
//         if (~IR && ~IPS_M_L && ~IPS_M && IPS_M_R && IPS_R_R) state <= S3;
       if (~IPS_M_L && IPS_M && IPS_M_R) state <= S3;
     else    state <= S2;
   end
   S3: begin //alternate path
//         if (~IPS_M_L && ~IPS_M && ~IPS_M_R && ~IPS_R_R) state <= S4;
//           if (~IPS_M_L && ~IPS_M && ~IPS_M_R) state <= S4;
         if (~IPS_M && ~IPS_M_R) state <= S4;
       else    state <= S3;
   end
   S4: begin
//         if (IPS_R_R && ~IPS_M_R) state <= S1;
         if (~IPS_M_L && IPS_M && IPS_M_R) state <= S5;
       else    state <= S4;
   end
   S5: begin
       if (~IPS_M_L || ~IPS_M || ~IPS_M_R) state <= S5;
       else state <= S5;
   end
//     S6: begin
//     if (~IPS_M) state <= S6;
//     else state <= S6;
//     end
   default: begin // default case to avoid error for unreachable state
     state = S0; // always go to state 0
   end
 endcase
```

Figure 13: State Machine Code for Final Project [9].

The alternative path consisted of two 90 degree left turns. Once the rover is in the state after making the initial right turn the stat declares that if either the left IPS or the middle and left IPS make contact the rover will turn left. Once the rover completes both turns the next turn is the convergent path to get back on the normal path. To dio this a new state is created that says if the middle, left and right IPS sensors are active the rover will turn right. Since the remaining path has no occurrence of all three sensors being active at the same time this state proves to be valid.



Load Percentage (%)	Motor A Voltage (V)	Motor B Voltage (V)
0%	0	0
25%	5.56	5.5
50%	7.01	6.95
75%	7.52	7.55
85%	7.71	7.73



Load Percentage (%)	Motor A Voltage (V)	Motor B Voltage (V)
0%	0 V	0 V
25%	5.5 V	5.3 V
50%	7.02 V	6.8 V
75%	7.6 V	7.46 V
85%	7.74 V	7.59 V

Figure 14: Reverse and Forward Voltages relation to Load Percentage (%).

The graphs [Figure 14] show that the motors can have different speeds even at the same set speed. This graph explains why three IPS sensors are needed in the middle. The track can be a straight line, however, since the motor speeds vary the rover can eventually go off course. The IPS in the middle is needed in order to maintain contact with the track while the outer two keep the rover moving between the lines. For example, when the right IPS is activated the algorithm identifies that the rover is moving towards the left of the track and needs to move towards the right to keep following the line.

5. Demo Day Troubleshooting

When attempting to solve some remaining issues, the main problem involved the rover not able to make 90 degree turns. More time and effort were put into troubleshooting this issue. The overall code was programmed correctly, however, after visualizing the process that occurs during turns it was discovered that the middle IPS had to be moved slightly forward for the correct sequence of IPS to move it into a specific state. The last minute solution [Figure 15] to this problem was by using tension from zip ties to pull the middle IPS slightly forward.

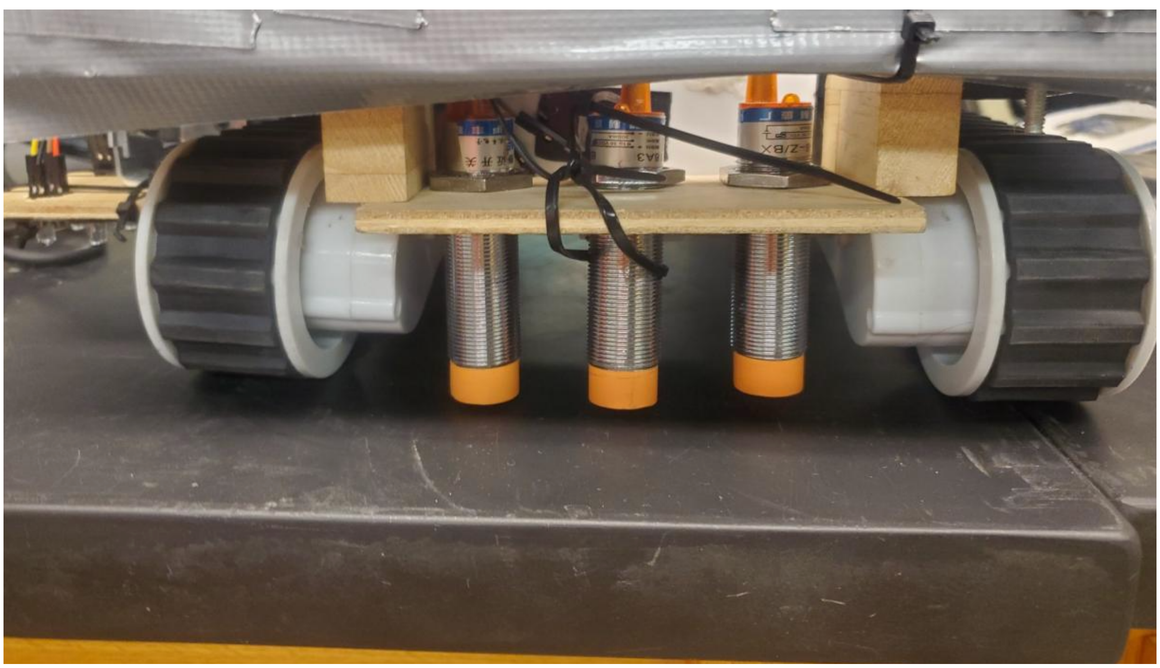


Figure 15: IPS Positioning Troubleshooting

6. Ethics

Some ethical concerns that had occured within the project was the issue of conflict of interest. Since the individuals involved had different assignments within the project there were different viewpoints of how the hardware implementation would be carried out. Overall the conflict of interest on different aspects of the project proved to be one of the major ethical issues.

7. Safety

In terms of safety, the lab room requires proper attire in order to operate within the lab. An example was set when the Chair of the Electrical and Computer Engineering removed a student for not wearing proper attire. Safety was also used within the wood shop by using proper handwear in order to avoid splinters. Proper eyewear was also used in order to keep debris out of the eyes. Utilizing the table saw and drill press required different safety requirements in order to use.

8. Conclusion

Up to this point the utilization of different components and the understanding of each piece is important in order to implement them together. The hardware component plays an important role in the software and vice versa. The main project required all these components to work together in order to carry out the main project description. The final rover [Figure 15] has all of the hardware components connected along with the color sensor mount on the side,

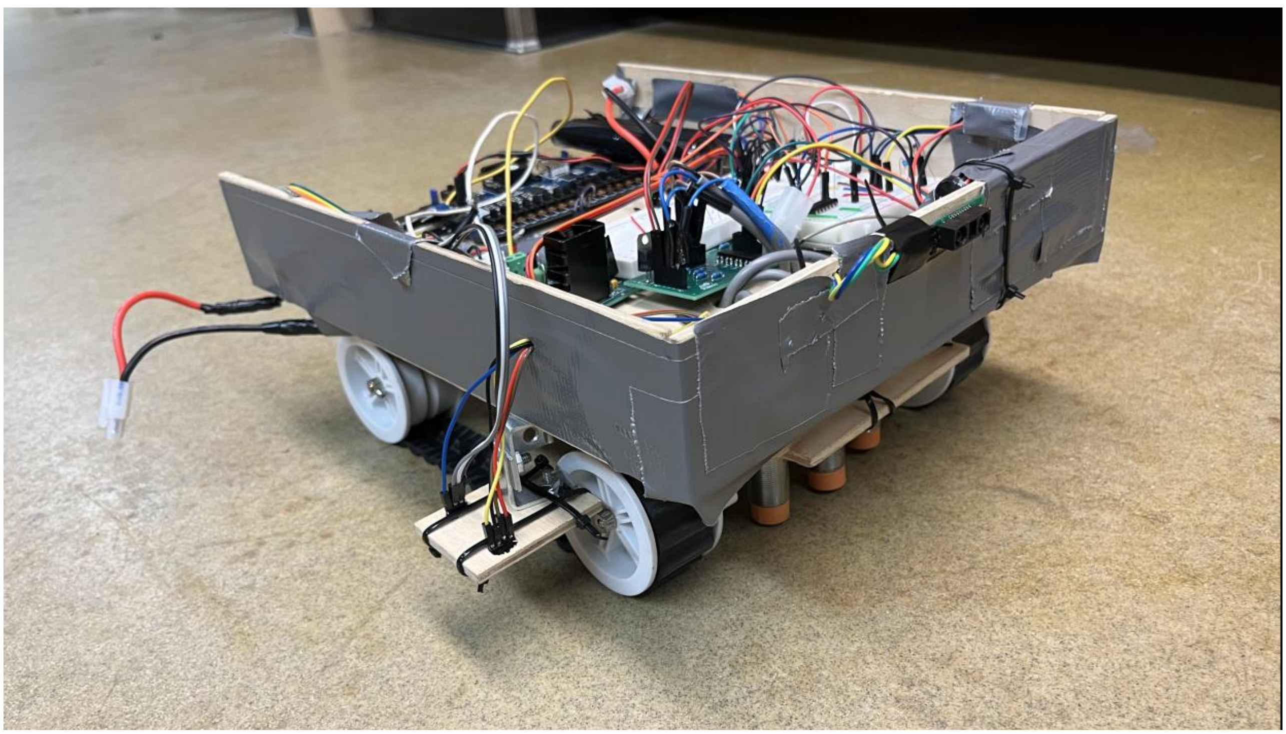


Figure 16: Final Design with IPS IR and Color Sensor Mounted.

References

1. *Basys 3 FPGA Board Reference Manual*. Diligent. (2016, April 8). Retrieved September 19, 2021, from https://digilent.com/reference/_media/basys3:basys3_rm.pdf.

2. *[FPGA Tutorial] seven-segment LED display On Basys 3 Fpga*. FPGA Projects, Verilog Projects, VHDL Projects - FPGA4student.com. (n.d.). Retrieved September 20, 2021, from https://www.fpga4student.com/2017/09/seven-segment-led-display-controller-basys3-fpga.html.

3. *How to use the L298N Dual H-Bridge Motor Driver*. Banana Robotics. (n.d.). Retrieved September 20, 2021, from https://www.bananarobotics.com/shop/How-to-use-the-L298N-Dual-H-Bridge-Motor-Driver.

4. Dahl, Ø. N. (2020, June 25). *What is an H-Bridge?* Build Electronic Circuits. Retrieved September 21, 2021, from https://www.build-electronic-circuits.com/h-bridge/.

5. Cook, J. S. (2021, June 14). *NPN vs PNP: What's the difference?* Arrow.com. Retrieved October 24, 2021, from

https://www.arrow.com/en/research-and-events/articles/npn-vs-pnp-in-circuit-design-and-industrial-controls.

6. *What is an Optocoupler and How it Works*. Jameco. (n.d.). Retrieved October 24, 2021, from https://www.jameco.com/Jameco/workshop/Howitworks/what-is-an-optocoupler-and-how-it-works.html.

7. *Op-amp comparator* . Basic Electronics Tutorials. (2021, March 19). Retrieved October 24, 2021, from https://www.electronics-tutorials.ws/opamp/op-amp-comparator.html.

Figures Reference:

1. Figure 1: J. Ramirez *State Machine Flowchart.* 2021.
2. Figure 3: *LM340/LM78XX Series 3-Terminal Positive Regulators*. DataSheet4U. (2006, July). Retrieved September 20, 2021, from https://datasheet4u.com/datasheet-pdf/NationalSemiconductor/LM340/pdf.php?id=49663
3. Figure 4: Dahl, Ø. N. (2020, June 25). *What is an H-Bridge?* Build Electronic Circuits. Retrieved September 21, 2021, from https://www.build-electronic-circuits.com/h-bridge/.
4. Figure 5: Panhorst, James. (2021). *IPS Schematic*. Retrieved October 24, 2021.

5. Figure 7: J. Panhorst. *IPS PCB*. 2021.
6. Figure 8: *PS2501-1, PS2501-4, PS2501L-1, PS2501L-4 data sheet*. (n.d.). Retrieved October 24, 2021, from https://kr.mouser.com/datasheet/2/698/REN_r08ds0202ej0100_ps2501_1_4_DST_20201225-2065566.pdf.
7. Figure 9: LM339 Reference Manual. Semiconductor Industries (2014, October) Retrieved September 20, 2021, from https://www.onsemi.com/products/signal-conditioning-control/amplifiers-comparators/comparators/lm339s.
8. Figure 10: J. Panhorst. *IR Schematic*. 2021.
9. Figure 13: J. Ramirez. *State Machine Code*. 2021.

Appendix A

Gantt Chart

Main Project							9/20/21	9/27/21	10/4/21	10/11/21	10/18/21	10/25/21	11/1/21	11/8/21	11/15/21	11/21/21
Task Name	Done	Josh	James	Jibin	Alex	Location					Interim Presentaion	Interim Report Due	PCB Deadline			
Mini Project	100%	xx	xx	xx	xx	Lab										
EagleCAD Circuit Design	100%	xx	xx	xx	xx	Lab/Internet										
Hardware																
IPS Sensor																
Research	100%	xx	xx	xx	xx	Internet										
Obtain Part	100%	xx				Lab										
Implementation	100%		xx	xx		Lab										
Connectivity	100%		xx	xx		Lab										
IR Sensor																
Research	100%	xx	xx	xx	xx	Internet										
Obtain Part	100%	xx				Lab										
Implementation	100%		xx	xx		Lab										
Connectivity	100%		xx	xx		Lab										
Light to Frequency Convertor																
Research	100%	xx	xx	xx	xx	Internet										
Obtain Part	100%	xx				Lab										
Implementation	66%		xx	xx		Lab										
Connectivity	75%		xx	xx		Lab										
Software																
IPS Sensor																
Research Software Implementation	100%	xx	xx	xx	xx	Internet										
Create Code Implementation	100%	xx			xx	Lab										
Test and Complete Code	100%	xx			xx	Lab										
IR Sensor																
Research Software Implementation	100%	xx	xx	xx	xx	Internet										
Create Code Implementation	100%	xx			xx	Lab										
Test and Complete Code	100%	xx			xx	Lab										
Light to Frequency Convertor																
Research Software Implementation	100%	xx	xx	xx	xx	Internet										
Create Code Implementation	33%	xx			xx	Lab										
Test and Complete Code	50%	xx			xx	Lab										
Final Project Completion																
Finish Circuit Connection to Rover-5	66%	xx	xx	xx	xx	Lab										
Finish and Compile all Software	75%	xx	xx	xx	xx	Lab										
Test Software and Hardware Together	0%	xx	xx	xx	xx	Lab										

Appendix B

Budget

Project Lab 1	Current Totals			Estimated Total				Start Date	8/23/21
Direct Labor	Hourly Rate	Hours	Total Pay	Hourly Rate	Hours	Total Pay		**Today's Date**	11/29/21
James P.	15	156	$2,340.00	15	150	$2,250.00		**End Date**	11/29/21
Josh R.	15	158	$2,370.00	15	150	$2,250.00			
Alex H.	15	120	$1,800.00	15	150	$2,250.00			
Jibin M.	15	166	$2,490.00	15	150	$2,250.00			
Labor Subtotal		Subtotal:	$9,000.00		Subtotal:	$9,000.00			
Labor Overhead	Rate:	50%	$4,500.00	Rate:	50%	$4,500.00			
Total Labor			$13,500.00			$13,500.00			
Contract Labor	Hourly Rate	Hours	Total Pay	Hourly Rate	Hours	Total Pay			
Lab Assistant	$40.00	3	$120.00	$40.00	5	$200.00			
Total Contract Labor			$120.00			$200.00			
Material Costs	Quantity	Value	Total Cost						
Breadboard	3	$5.00	$15.00						
LM339 Comparator	2	$0.50	$1.00						
Voltage Regulator	5	$0.75	$3.75						
Jumper Wire Pack	2	$2.00	$4.00						
4 Channel Opto-Coupler	1	$1.38	$1.38						
Plywood	2	$8.00	$16.00						
Resistor	13	$0.13	$1.69						
6x1 Female Pin Headers	1	$0.52	$0.52						
2x5 Female Pin Headers	2	$0.49	$0.98						
Fuse Holder	1	$5.00	$5.00						
2A Fuse	5	$1.00	$5.00						
Total Material Costs			$54.32						
Equipment Rental Costs	Value	Rental Rate	Rental Cost	Value	Rental Rate	Rental Cost			
Oscilloscope	$5,300.00	0.20%	$1,038.80	$5,300.00	0.20%	$1,038.80			
Rover 5	$60.00	0.20%	$11.76	$60.00	0.20%	$11.76			
Function Generator	$16.00	0.20%	$3.14	$16.00	0.20%	$3.14			
Multi-Meter	$149.00	0.20%	$29.20	$149.00	0.20%	$29.20			
Battery Pack	$17.00	0.20%	$3.33	$17.00	0.20%	$3.33			
Basys 3 Artix-7	$150.00	0.20%	$29.40	$958.00	0.20%	$187.77			
IPS Sensor	$8.00	0.20%	$1.57	$8.00	0.20%	$1.57			
IR Sensor	$13.29	0.20%	$2.60	$13.29	0.20%	$2.60			
Color Light to Frequency Convertor	$4.00	0.20%	$0.78	$4.00	0.20%	$0.78			
Power Supply	$1,700.00	0.20%	$333.20	$1,700.00	0.20%	$333.20			
Total Rental Costs			$1,453.79			$1,612.16			
[illegible]			[illegible]			[illegible]			